**7-3 Project Two**

Jasmine Zeng

Southern New Hampshire University

CS 370: Current/Emerging Trends in Computer Science

Professor Travis Smith

**🧠 Analyze the differences between human and machine approaches to solving problems.**

Describe the steps a human being would take to solve this maze.

In a game, a person learns through rules and rewards. In the beginning of every “state", the person would read the map and make their move according to the rules. Their action would be taken to the one that reaps the highest reward, or helps them become one step closer to achieving it. In multiple trials, the person will learn which moves are “good” and which moves are “bad” in order to win the game.

Describe the steps your intelligent agent is taking to solve this pathfinding problem.

The intelligent agent learns through reinforcement learning. This means that they take in data input from the board, and make their moves accordingly. However, there is not much respect to rules, moreso it is a game of multiple trials and errors. Eventually, the intelligent agent learns which patterns help them maximize rewards. As the intelligent agent continues learning, it is assumed they are becoming increasingly precise at figuring out how to win the game.

What are the similarities and differences between these two approaches?

People learn through a priori, while intelligent agents learn through a posteriori. This means that people are influenced by the rules of the game when they are making decisions. Meanwhile, intelligent agents will execute multiple trials of the game and learn after the fact. Neither approach is wrong. However, intelligent agents tend to have a more efficient learning pattern in the long run since they attempt all possible moves and recognize patterns from it.

**🧠 Assess the purpose of the intelligent agent in pathfinding.**

What is the difference between exploitation and exploration? What is the ideal proportion of exploitation and exploration for this pathfinding problem? Explain your reasoning.

Exploitation is finding shortcuts, while exploration is attempting all possible moves within an environment. For intelligent agents, it is ideal for them to explore a majority of the time. This is because they can find all possible outcomes of the game and respond accordingly when they are deployed. Exploring allows the intelligent agent to expect certain outcomes and know how to navigate through them. Eventually, when the intelligent agent becomes knowledgeable enough, then they can begin exploiting.

How can reinforcement learning help to determine the path to the goal (the treasure) by the agent (the pirate)?

Reinforcement learning helps the intelligent agent learn from their mistakes. As mentioned in the previous question, if an intelligent agent learns all possible outcomes and picks up on patterns, it will be capable of handling all possibilities in the game. Intelligent agents are excellent at running several trials and learning from them. This way, they can determine the best, most statistically favorable move when approached by any game setup. Humans on the other hand may not have the time or ability to become this precise through pattern recognition and reinforcement learning. Many intelligent game agents today are even finding unprecedented patterns that even human experts have not recognized or found.

**🧠 Evaluate the use of algorithms to solve complex problems.**

How did you implement deep Q-learning using neural networks for this game?

I used deep Q-learning neural networks, or DQN, to help the intelligent agent find action-state pairs. With each action they took, they noted whether it resulted in a reward or a punishment. After several epochs, the DQN was able to help the intelligent agent find shortcuts to those explorations, or exploits. The agent began winning after consequent trials of the treasure hunt game. However, when running my algorithm, it took quite a bit and I actually was not able to finish its learning even after running the program for 12 hours. I think in the future I will work on better decision branching which may help the intelligent agent learn more efficiently.